CLUBE

Dezembro/83

N.º 15

Coordenadores: MARIA IRENE • ALEXANDRE SOUSA • J. MAGALHÃES • AV. BOAVISTA, 832-2.º T. • TELEF. 65127 • 4100 PORTO

NESTE NÚMERO

Edição: Clube Z80

Fotocomposição: Fotomecânica Mabreu/Porto

Impressão: Gráfica Firmeza/Porto

Tiragem: 500 exemplares, Dezembro 1983

EDITORIAL

DEZEMBRO/83

Neste findar de 83 é tentador deitar uma olhadela ao que fizemos e ao que pretendemos fazer.
Em termos da n/ revista (ou jornal) passamos à impressão tipográfica e talvez tenhamos produzido um trabalho mais sistematizado. Tivemos a excepcional colaboração de Fernando Preces e pensamos ter conseguido agarrar a viragem do ZX 81 para o Spectrum.
É altura de perguntar: quantos possuidores do ZX 81 possuem agora o Spectrum? Venderam-se em Dezembro entre 5 000 a 10 000 Spectrum's. Se o n/ associado está registado como possuidor do ZX 81 e trocou de máquina durante este ano (posteriormente ao n/ inquérito) é importante nesta altura conhecer o grau de interesse que ainda existe para programas do ZX 81/TS 1 000.

———•———

O número de associados é de 204, ainda não suficiente para manter a impressão da revista mensal e suportar as despesas decorrentes do funcionamento dos serviços a prestar aos sócios (fotocópias, livros, programas, CTT's, etc.).
É importante fazer um esforço para aumentar o número de associados.

———•———

O número de jogos que existem no mercado, para usar com o Spectrum, atingiu já a centena. Gostaríamos que os n/ associados nos escrevessem, comentando os jogos e explicando o seu funcionamento, dado que, com o florescimento da «pirataria», os jogos chegam muitas vezes às mãos dos utilizadores sem instruções capazes.

———•———

Continuamos atentos às mensagens que nos chegam a solicitar programas para o **Apple** ou **New Brain** ou ainda versando problemas de Engenharia Civil ou outras áreas. Não se trata de esquecimento nosso — é uma questão de oportunidade. Quando isso for possível tentaremos sempre que a voz de cada um tenha sentido nas n/ páginas.

———•———

Finalmente, aqui vão as n/ desculpas pelo grande atraso com que este número vos chega às mãos.

Um grande abraço para todos
Alexandre Sousa/J. Magalhães
Isabel Cristina/Maria Irene

Rui Carvalho/Barreiro, colocou uma questão que nos parece importante para muitos utilizadores do SPECTRUM. (Atenção: Existe uma tendência dos constructores para abandonar o Basic) — QUAIS AS ALTERNATIVAS PARA O BASIC? QUAIS AS VANTAGENS DO "PASCAL" OU "FORTH"?
Neste momento existem COMPILADORES quer para a linguagem PASCAL, quer para FORTH, em cassete, para os possuidores do SPECTRUM.

FORTH oferece uma selecção de estruturas de controlo do programa, de modo a permitir que a execução seja condicional e interactiva.
Outro aspecto fundamental é que podem ser adicionadas novas especificações a uma "standard" já existente.
Esta linguagem tem já 10 anos de existência e começou por ser usada na resolução de problemas científicos críticos e em aplicações indústriais. Em termos do entusiasta da programação, os aplausos vão para a eficiência com que a linguagem cumpre a sua função.

FORTH é simultaneamente:

- Linguagem de alto nivel
- Linguagem "Assembly"
- Sistema Operatório
- Conjunto de ferramentas de desenvolvimento
- Filosofia de projectos de programação

Como LINGUAGEM, FORTH começa com um poderoso conjunto de comandos que proporcionam os mecanismos que permitem ao programador definir os seu próprios comandos.
A codificação, ao mais alto nível, parece uma descrição em lingua inglesa.
É uma linguagem que pode ser usada para a resolução de problemas.

Exemplo:

```
MACHO?
IF 1 HOMEM + !
ELSE 1 MULHER + !
THEN
```

Se o resultado de MACHO? é verdadeiro, a execução continua desde ELSE ou THEN, conforme o que estiver em primeiro lugar.

(Cont. no próximo número)

INTRODUÇÃO À LINGUAGEM MÁQUINA

ZX81

FINAL DO 1.º CAPÍTULO — (Secção de Jogos) — Pequena recapitulação da matéria estudada

(Continuação)

JOGO 1 — (Rotina SCROLL *ESTE-OESTE*, e pequena rotina de *Pesquisa de colisão*)

```
   1 REM (Reservar 28 caracteres)
   8 GOSUB 1000
  10 LET S = 0
  20 LET A = CODE "<" (símbolo menor que)
  30 LET B = A
  50 LET Z = S
 100 FOR D = S TO A * A
 110 PRINT AT B, Z;
 120 IF USR 16514 = A THEN GOTO 300
 130 PRINT "O" (código 52)
 140 IF INKEY$ = "F" THEN GOSUB 500
 190 LET B = B — (INKEY$ = "7" AND B)
     + (INKEY$ = "6" AND B < A)
 200 PRINT AT RND * A, 17; "<"
 210 LET C = USR 16521
 220 NEXT D
 300 PRINT "▦"; S; "PONTOS" (código gráf. 8)
 310 FOR T = 0 TO 300
 320 NEXT T
 325 CLS
 330 PRINT AT 12,0; "QUER CONTINUAR? (Diga S
     ou N)."
 340 INPUT L$
 345 CLS
 350 IF L$ = "S" THEN GOTO 10
 355 PRINT AT 12,0; "ATE À PRÓXIMA E . . . OBRI-
     GADA."; W
 500 FOR C = Z TO PI (símbolo π)
 510 IF USR 16514 = A THEN LET S = S + 1
 520 PRINT "-";
 530 NEXT C
 540 PRINT AT B, Z; "O" (Código 52 seguido de 4
     espaços)
 550 RETURN
1000 PRINT AT 1,5; "BATALHA NO ESPAÇO"
1010 PRINT,, "A SUA NAVE É ATACADA POR
     CENTE-"
1020 PRINT,, "NAS DE PEQUENAS NAVES, CUJO
     FO-"
1030 PRINT,, "GO É INOFENSIVO MAS, PERI-
     GOSAS"
1040 PRINT,, "AO CHOQUE."
1050 PRINT,, "A T 7 DESVIA A NAVE PARA CIMA
     E A"
1060 PRINT,, "T 6 PARA BAIXO. A TF DISPARA."
1070 PRINT,, "BOA SORTE . . ."
1080 PRINT,, "PRIMA" "N/L" "PARA COMEÇAR."
1090 INPUT L$
1095 CLS
1099 RETURN
2000 SAVE "JOGO 1"
2010 RUN
```

NOTA: Este jogo pode correr numa máquina de 1 K RAM, se forem eliminadas as linhas (8, 310 a 360 e 1000 a 1099) acrescentando na linha 300 (; W).

Listagem do código máquina com início em 16514

(42, 14, 64, 78, 6, 0, 201, 1, 117, 23, 3, 42, 16, 64, 22, 0, 43, 126, 185, 40, 4, 114, 87, 24, 247, 16, 243, 201.)

Uma grande parte do programa escrito em Basic, incluindo a movimentação da nave e os seus disparos, a introdução de outros movimentos como acelerar ou travar, o aparecimento de outros obstáculos, etc., podem ser transformados em rotinas CM, que farão o jogo mais rápido e aliciante.

Em código máquina a movimentação por tecla é inquirida pela consulta cíclica a uma variável do sistema chamada Last K (última tecla premida) que depois é enviada por uma instrução CALL para a rotina de descodificação do teclado (na ROM), e o retorno entregue a uma rotina de movimento que soma ou subtrai o número 33 (salto de uma linha) à posição PRINT no ficheiro de projecção, ocupada pela nave. Quanto aos disparos, se a tecla F foi premida é comparada com o seu código por uma instrução CP, N e a informação conduzida com uma instrução JRZ ou CALL para a rotina de disparos.

Não é difícil, e o leitor mais afoito ou mais experiente nestas andanças de programação em código máquina pode tentar transformar algumas dessas instruções do Basic. Quanto ao leitor que não possua ainda essa experiência, terá de ser paciente e aguardar o estudo dessas rotinas, para melhorar este e outros programas.

A segunda rotina máquina deste jogo, que começa em 16521, é já nossa conhecida. Ela executa o Scroll Este-Oeste.

A primeira, com começo em 16514, é uma novidade de que ainda não tínhamos falado.

É muito pequena e apenas substitui 3 instruções do Basic e as respectivas operações aritméticas.

As linhas Basic 120 e 510, antes de ser introduzida a rotina máquina eram assim:

```
120 IF PEEK (PEEK 16398 + 256 * PEEK 16399) = A
    THEN GOTO 300
```

```
510 IF PEEK (PEEK 16398 + 256 * PEEK 16399) = A
    THEN LET S = S + 1
```

Em termos de programa elas executam rigorosamente o mesmo trabalho só que a diferença em tempo de execução é de 3 para 1.

Interpretação da rotina em CM:

42	LD, HL, (NN)	carrega HL com o endereço da posição PRINT no ficheiro de projecção
14 64	16398	
78	LD C, (HL)	carrega C com o conteúdo de HL
6 0	LD B, N 0	coloca B a 0 para o retorno
201	Ret	retorno ao Basic

Para uma melhor compreensão vamos interpretar o que está programado nas linhas Basic 120 e 510.

Linha 120

Se uma pequena nave (<) for ocupar o espaço da nave (O), temos uma colisão, e com GOTO 300, a explosão e a pontuação até ao momento, terminando o jogo. Por outras palavras: se o conteúdo de HL for igual ao código 19 (<) então há colisão.

Linha 510

Ao ser premida a tecla F, se uma pequena nave for ocupar o espaço destinado ao disparo (–), *nova posição PRINT do ficheiro*. A nave (<) é eliminada e ganha-se o ponto, com LET S = S + 1.

Mais para a frente, este jogo sofrerá as transformações compatíveis ao desenvolvimento do estudo da linguagem máquina efectuado, até atingir algumas das potencialidades de um jogo como o GROUND ATTACK.

Para habituar o leitor ao estudo e montagem dum programa, sobre o seu organigrama, um dos assuntos a ser tratado no próximo capítulo, acrescenta-se aqui o diagrama bloco deste jogo.

(Cont. no próximo número)

SIMULAÇÃO

PARTE III/DEZ. 83

Referimos anteriormente que a nossa próxima etapa é colocar o relógio no início do dia.
A linha de espera dos clientes será indicada pela variável L e será inicializada com o valor zero. A variável representativa das perdas de caixa será PC e o total de vendas para o dia será indicado pela variável TV; ambas serão colocadas em zero.

```
300 LET HT = 7 : LET MT = 0
310 LET L = 0 : LET PC = 0 : LET TV = 0
```

No início de cada hora, o programa designa o número de clientes. Para a hora "J", representará a chegada do cliente A(J). A cada cliente será atribuído um tempo de chegada em minutos (após a hora) e o computador irá escolher o tempo de chegada, usando o gerador de números aleatórios (RANDOM).
Na ausência de qualquer outra informação, assumimos que o cliente vai sair aleatoriamente uns minutos após a hora. Vamos ver como isto se processa em termos do computador:
No início de cada hora simulada, lançamos uma tabela, D(T), com 15 entradas, uma por cada período de 4 minutos (dentro de cada hora).
Esta tabela ou "array" deverá indicar quantos clientes chegam em cada intervalo de 4 minutos. Por exemplo, se D(10) = 4, então quatro clientes chegam entre o minuto 36 e o minuto 40 (dessa hora), ou seja, no décimo intervalo do tempo dessa hora.
O programa vai situar aleatoriamente cada um dos clientes A(J), num intervalo de 4 minutos, usando o gerador de números aleatórios. O nosso programa deverá testar o tempo para o início da hora. Isto será feito pela determinação de MT igual a zero (linha 410). No caso disso suceder, irá para a linha 1200, onde situa o cliente que chegou na hora exacta do minuto zero.

```
  11 DIM D(15)
 ......
 410 IF MT = O THEN GOSUB 1200
 ......
1200 FOR S = 1 TO 15
1220 NEXT S
1230 FOR I = 1 TO A(HT)
1240 LET X = INT (15 * RND) + 1
1250 LET D(X)=D(X)+1
1260 NEXT I
1270 RETURN
```

O programa irá agora progredir ao longo da hora simulada em segmentos de 4 minutos.

Para o segmento de ordem T, ele origina que o cliente D(T) chega ao bar.
Vamos assumir que metade destes clientes irão ficar e outra metade entra e sai logo.
O computador irá dar uma olhadela a cada um destes clientes e decidirá quais os que ficam e quais os que irão sair. Se o cliente vai ficar, será adicionado à linha do cliente que permaneceu no bar. Se o cliente entra e sai, o computador adiciona à variável PC o valor 57$50, que significa um valor de caixa perdido. Quando o cliente permanece na linha, para ser atendido, será acrescentado o valor 57$50 à variável TV (total de vendas). Finalmente, o tempo será actualizado e o mesmo procedimento será repetido no próximo segmento de 4 minutos.
Posto isto, teremos a finalidade desta simulação — analisar se este negócio exigirá ou não a admissão de outra pessoa, ou mesmo se deveremos mudar de negócio.
Vamos então ver a parte do programa onde é executada esta análise.
A linha 720 vai fazer o teste do tempo para o final do dia (HT = 19).
Quando isto se verifica, o progama vai para a linha 1500, onde o final do dia estatístico vai ser remetido para o ecran.

```
420 LET T = MT/4 + 1
430 FOR J = 1 TO D(T)
440 LET C = INT (2 * RND) + 1
450 IF C = 1 THEN GOTO 500
455 GOTO 600
490 REM 500-560 SAÍDA CLIENTE
500 IF RND > D(L) THEN GOTO 550
510 LET PC = PC + 57.50
520 GOTO 690
550 LET L = L + 1 : REM CLIENTE SAI
560 GOTO 690
590 REM 600-660 CLIENTE FICA
600 IF RND > P(L) THEN GOTO 640
610 LET PC = PC + 57.50 (CLIENTE PERDIDO)
620 GOTO 690
640 LET L = L + 1
690 NEXT J
700 IF L = O THEN GOTO 710
702 LET L = L - 1
704 LET TV = TV + 57.50 (CLIENTE GANHO)
710 GOSUB 1000 (ACTUALIZAR O TEMPO)
720 IF HT = 19 THEN GOTO 1500
725 GOTO 800
730 REM HT = 19 FINAL DO DIA
800 GOTO 410 (PRÓXIMO SEGMENTO DE
    4 MINUTOS)
```

```
1500 PRINT "FINAL DO DIA ESTATÍSTICO"
1510 PRINT "VALOR DE VENDAS PERDIDO",
     PC + PC * 57.50
1520 PRINT "VALOR REALIZADO EM CAIXA", TV
1530 PRINT "LINHA FECHADA AO CLIENTE N.º", L
2000 END
```

O nosso programa simula a actividade de um simples dia. Para termos uma média estatística, em relação a um certo número de dias, teremos de repetir o programa.
Vamos escolher arbitrariamente 10 dias.
A variável D1 vai registar esse número de dias. A variável TP regista o valor de negócio perdido e a variável T6 representa o total ganho. Estas duas variáveis serão actualizadas no final de cada dia.
O "dia número..." será representado por E e a mudança do dia será controlada pelo ciclo das linhas 290 e 1700:

```
┌──── 290 FOR E=1 TO 10 ──┐
└──── 1700 NEXT E ────────┘
```

Estatisticamente, iremos calcular a média das perdas PC, o total de vendas/dias e o fecho da linha diária L. Teremos em conta os valores destas variáveis para todos os dias; estão representadas por L1, C1 e CL, respectivamente.

```
1500 LET L1 = PC + L1 + L * 57.50
1510 LET C1 = TV + C1
1520 LET CL = L + CL
```

As linhas 1800 a 1850 calculam as médias de L1, C1 e CL e mostram os seus resultados.

(Cont. no próximo número)

ENCICLOPÉDIA DA LINGUAGEM BASIC

ASC

A função ASC converte um caracter ou variável alfanumérica (string) no seu valor numérico decimal (código ASCII).

Por exemplo:

Print ASC ("A") apresentará o valor 65 no ecran, dado que esse é o seu valor em código ASCII.
Se pedir: Print ASC (A$), obterá o código do primeiro caracter da variável A$.

PROGRAMA 1

```
10 PRINT "O CÓDIGO ASCII PARA A LETRA
   'A' SERÁ = ";
20 PRINT ASC ("A")
30 IF ASC ("A") = 65 THEN 70
40 PRINT "TESTE FALHADO POR 'ASC'"
60 GOTO 99
70 PRINT "TESTE OK! PARA 'ASC'"
99 END
```

PROGRAMA 2

```
10 PRINT "ENTRADA PARA QUALQUER NÚ-
   MERO, LETRA OU CARACTER"
20 INPUT A$
30 PRINT "O CÓDIGO ASCII PARA ";A$;"
   E = ";ASC(A$)
40 GOTO 20
```

ALGUNS COMPUTADORES QUE INCORPORAM A FUNÇÃO 'ASC' PODEM ACEITAR STRINGS SUPERIORES A UM CARACTER, MAS SOMENTE O PRIMEIRO CARACTER SERÁ CONVERTIDO NO RESPECTIVO CÓDIGO.

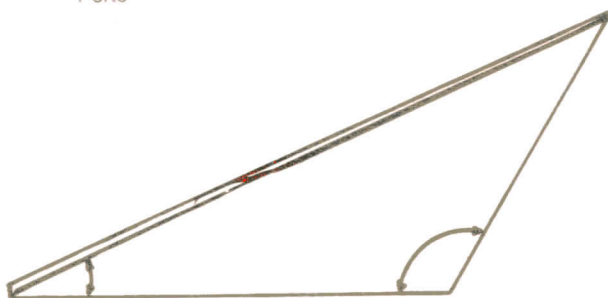
OBSERVAÇÃO

NAS MÁQUINAS SINCLAIR, ESTA FUNÇÃO TEM O NOME DE... CODE... PELO QUE PODE TESTAR O TEXTO DESTE MÊS, USANDO A FUNÇÃO 'CODE' EM VEZ DE 'ASC'.

(Cont. no próximo número)

RESOLUÇÃO DE TRIÂNGULOS

SPECTRUM/ZX81/TS 1000

Adapt. ALEXANDRE SOUSA
Porto

```
PROGRAMA PARA RESOLUCAO DE TRIAN
GULOS
PODE SER ADAPTADO SEM DIFICULDA-
DE A QUALQUER MAQUINA
FOI ESCRITO PARA O SINCLAIR   SP
ECTRUM

PARA OS POSSUIDORES DO ZX81
CHAMA-SE A ATENCAO P/ AS LINHAS
TIPO....INPUT'LADO='

PODEM SER SUBSTITUIDAS POR:

PRINT'LADO='
INPUT L(1)

TAMBEM DEVEM NOTAR QUE O SIM-
BOLO  ↑  DEVE SER SUBSTITUIDO
POR  **

ESTE PROGRAMA FOI ADAPTADO DE
OUTRO EXISTENTE P/APPLE II

EXEMPLO : PROBLEMA TIPO 2

LADO=8.76  ANGULO=90

LADO=8.76

RESULTADOS
LADO 1=12.388
ANGULO OPOSTO=90 GRAUS
LADO 2=8.76
ANGULO OPOSTO=45 GRAUS
LADO 3=8.76
ANGULO OPOSTO=45 GRAUS

EXEMPLO :PROBLEMA TIPO 5

LADO=10  LADO=13 LADO=14

RESULTADOS
LADO 1=10
ANGULO OPOSTO=43.279 GRAUS
LADO 2=13
ANGULO OPOSTO=63.027 GRAUS
LADO 3=14
ANGULO OPOSTO=73.693 GRAUS
```

```
   1 REM Programa Educacional
   2 REM p/construir um triangul
o
   3 REM dadas as partes conheci
das
   4 REM o programa calcula as p
artes desconhecidas
 100 CLS
 110 DIM A(3)
 115 DIM L(3)
 120 REM A=ANGULO L=LADO
 130 PRINT AT 10,0;"1- A L A"
 140 PRINT AT 12,0;"2- L A L"
 150 PRINT AT 14,0;"3- A A L"
 160 PRINT AT 16,0;"4- L L A"
 170 PRINT AT 18,0;"5- L L L"
 180 PRINT AT 20,10;"ESCOLHA...?
"
 182 INPLT Y$
 183 IF Y$="1" THEN GO TO 1000
 184 IF Y$="2" THEN GO TO 190
 185 IF Y$="3" THEN GO TO 260
 186 IF Y$="4" THEN GO TO 300
 187 IF Y$="5" THEN GO TO 390
 188 GO TO 182
 190 REM 2
 191 CLS
 192 INPUT "LADO=";L(3)
 196 INPUT "1)ANGULO/GRAUS=";A
 197 GO SUB 2000
 198 LET A(1)=A
 199 INPUT "LADO=";L(2)
 210 LET L(1)=SQR (L(3)↑2+L(2)↑2
-2*L(3)*L(2)*COS (A(1)))
 220 LET A(2)=SIN (A(1))/L(1)*L(
2)
 230 LET A(2)=ATN (A(2)/SQR (-A(
2)*A(2)+1))
 240 LET A(3)=PI-A(1)-A(2)
 250 GO TO 440
 260 REM 3
 261 CLS
 262 INPUT "1)ANGULO=";A
 264 GO SUB 2000
 266 LET A(3)=A
 268 INPUT "2)ANGULO=";A
 270 GO SUB 2000
 272 LET A(2)=A
 274 INPUT "LADO=";L(3)
 276 LET A(1)=PI-A(2)-A(3)
 280 GO TO 1050
 300 REM 4
 301 CLS
 302 INPUT "1)LADO=";L(1)
 304 INPUT "2)LADO=";L(2)
 310 INPUT "ANGULO=";A(1)
 320 LET T=L(2)*SIN (A(1))
 330 IF L(1)<=T THEN GO TO 380
 340 LET L(3)=SQR (L(2)↑2-T↑2)
 350 IF L(1)<=T THEN GO TO 380
 360 LET Y=SQR (L(1)↑2-T↑2)
 370 LET L(3)=L(3)+Y
 380 GO TO 220
 390 REM 5
 391 CLS
 392 INPUT "1)LADO=";L(1)
 394 INPUT "2)LADO=";L(2)
 396 INPUT "3)LADO=";L(3)
 410 LET A(1)=(L(2)↑2+L(3)↑2-L(1
)↑2)/2/L(2)/L(3)
 420 LET A(1)=-ATN (A(1)/SQR (-A
(1)*A(1)+1))+1.5708
 430 GO TO 220
 440 CLS
```

```
442 PRINT "RESULTADOS"
443 FOR I=1 TO 3
444 IF A(I)<0 THEN GO TO 520
445 LET A(I)=A(I)*57.29578
446 PRINT "LADO ";I;"=";INT (L(
I)*1000+.5)/1000
448 PRINT "ANGULO OPOSTO=";INT
(A(I)*1000+.5)/1000;" GRAUS"
450 NEXT I
460 PRINT AT 20,0;"0.TECLA P/NO
VO PROBLEMA"
470 INPUT Y$
480 GO TO 100
520 CLS
522 PRINT "PROBLEMA SEM SOLUCAO
"
524 PAUSE 100
530 GO TO 100
```

```
1000 REM 1
1005 CLS
1010 INPUT "1)ANGULO/GRAUS=";A
1022 GO SUB 2000
1023 LET A(1)=A
1027 INPUT "LADO=";L(3)
1030 INPUT "2)ANGULO/GRAUS=";A
1034 GO SUB 2000
1036 LET A(2)=A
1040 LET A(3)=PI-A(1)-A(2)
1050 LET L(1)=L(3)*SIN (A(1))/SI
N (A(3))
1055 LET L(2)=L(3)*SIN (A(2))/SI
N (A(3))
1060 GO TO 440
2000 REM FACTOR CONV.GR--RAD
2010 LET A=A*.0174533
2020 RETURN
```

CALORIAS

SPECTRUM/ZX 81/TMS 1 000

CARLOS SILVA
Tomar

Este programa diz-nos o total de calorias/dia necessárias a um indivíduo, mediante a idade, sexo, peso, altura, horas de sono, horas de repouso e horas de trabalho diário.

```
IDADE=36
SEXO (M/F)...M
PESO...KG=62
ALTURA (CM)=164
HORAS DE SONO=7
HORAS DE REPOUSO=3
HORAS DE TRABALHO LIGEIRO=4
HORAS DE TRABALHO MEDIO=10
HORAS DE TRABALHO PESADO=0

TOTAL Kcal/dia =3385.4645

  1 REM CARLOS SILVA*TOMAR*
  5 LET C=.425
 10 LET D=.725
 20 LET E=.504
 22 LET F=.167
 24 LET G=.442
 26 LET H=.151
 30 PRINT "IDADE=";
 40 INPUT K
 42 PRINT K
 50 PRINT "SEXO (M/F)...";
 55 INPUT S$
 56 PRINT S$
 60 PRINT "PESO...KG=";
 65 INPUT I
 66 PRINT I
 70 PRINT "ALTURA (CM)=";
 75 INPUT J
 76 PRINT J
 80 IF S$="F" THEN GO TO 95
 85 LET M=E*I↑C*J↑D/K↑F
 90 GO TO 99
 95 LET M=G*I↑C*J↑D/K↑H
 99 PRINT "HORAS DE SONO=";
100 INPUT C
101 PRINT C
110 PRINT "HORAS DE REPOUSO=";
115 INPUT D
116 PRINT D
120 PRINT "HORAS DE TRABALHO LI
GEIRO=";
125 INPUT E
126 PRINT E
130 PRINT "HORAS DE TRABALHO ME
DIO=";
135 INPUT F
136 PRINT F
140 PRINT "HORAS DE TRABALHO PE
SADO=";
145 INPUT G
146 PRINT G
150 IF 24=C+D+E+F+G THEN GO TO
200
160 PRINT "ERRO DE CALCULO...SO
MAR DE NOVO AS HORAS"
170 INPUT S$
180 CLS
190 GO TO 99
200 PRINT AT 20,0;"TOTAL Kcal/d
ia =";INT C*M+(D+2*E+3*F+4*G)*I*
1.154
```

DESENHO

SPECTRUM/ZX 81/TMS 1000

CARLOS SILVA
Tomar

Este programa desenha um caracter que pode ser movimentado para cima ou baixo e para a esquerda ou direita. Para mudar de caracter, prima 0 (zero).

```
   1 REM CARLOS SILVA*TOMAR*
   5 REM "DESENHO"
  10 LET B=16
  20 LET A=11
  30 INPUT A$
  40 PRINT AT A,B;A$
  50 LET B$=INKEY$
  60 IF B$="" THEN GO TO 50
  70 IF B$="8" THEN LET B=B+1
  80 IF B$="5" THEN LET B=B-1
  90 IF B$="6" THEN LET A=A+1
 100 IF B$="7" THEN LET A=A-1
 110 IF B$="0" THEN GO TO 30
 120 GO TO 40
```

```
###################
                  #
                  #
                  #
                  #
                  #
                  #
                  #
                  #
                  #
#############     #
#                 #
#                 #
#                 #
#                 #
#                 #
#                 #
#                 #
#                 #
###################

ESTE DESENHO FOI FEITO C/MUITA S
IMPLICIDADE USANDO ESTE PROGRAMA
```


NOVOS PROGRAMAS

SPECTRUM

- **COOKIE** — Os ingredientes que o cozinheiro utiliza para fazer o seu bolo recusam-se a entrar na tigela e agridem o "cookie". Este atira-lhes com bombas de farinha. Mas há ainda o "Monstro do lixo" a atacá-lo.
- **SPACE WAR** — Simulação de uma batalha espacial.
- **ZOOM** — Você vai repelir ataques aéreos, terrestres e marítimos.
- **PSSST** — Usando insecticidas, o jardineiro tenta proteger a sua planta de ser atacada por pragas esfomeadas.
- **TRANZAM** — Você controla um bólide que atravessa o continente americano. Mas outros carros perseguem-no.
- **INVASÃO GALÁCTICA** — Para deter a invasão você controla 15 mísseis.
- **DROP A BRICK** — Destrua uma parede de tijolos com uma bola.
- **SILHOUETTE** — Desenhe qualquer figura e use a impressora para guardar uma cópia.
- **ARMAGEDON** — Destrua os satélites inimigos e evite a destruição das cidades, repelindo os misseis nucleares.
- **FRENZIE** — Você está num labirinto e se tocar nas paredes carregadas com 20 000 000 volts, pode ficar destruído — fuja dos robots assassinos! Para isso possui um "Laser" que dispara no sentido em que se desloca.
- **VIOLENT UNIVERSE** — Guerra no espaço.
- **ZIP ZAP** — Um robot vai explorar a possível colonização do universo. A sua energia provém de células de fuel e a defesa é feita com laser. Pode usar um escudo protector, activado por 10 unidades de energia. Quando passa de um sector para outro, tem um bónus de energia. O "Score" depende do sector e do número de inimigos abatidos.

Preço-programa 400$00

DESCONTO DE 20% para Sócios do Clube

FICHEIRO

SPECTRUM

Autor: RICARDO JORGE VAZ
Porto

O programa FICHEIRO tem capacidade de 96 discos. Depois de ter introduzido o programa no computador, aparecerá um MENU no ecran com 5 hipóteses:

1.ª — criar uma ficha;
2.ª — consultar (todo o ficheiro ou só uma ficha);
3.ª — alterar;
4.ª — verificar e gravar;
5.ª — imprimir.

Nota: Com ligeiras alterações, este programa pode adaptar-se ao ZX81 ou TMS1000.

```
   2 REM FICHEIRO
   5 LET CF=0
   8 PAPER 7: BORDER 7: INK 0: C
LS
  10 DIM C$(95,20)
  20 DIM E(95)
  30 DIM M$(95,10,17)
  40 DIM N$(95,10,17)
  50 LET D=9000
 100 REM menu
 105 CLS
 110 PRINT AT 0,14;"MENU"
 120 PRINT AT 3,10;"1-CRIAR"
 130 PRINT AT 6,10;"2-CONSULTAR"
 140 PRINT AT 9,10;"3-ALTERAR"
 150 PRINT AT 12,10;"4-GRAVAR E
VERIFICAR"
 160 PRINT AT 15,10;"5-IMPRIMIR"
 170 PRINT AT 18,5;"A SUA OPCAO"
 180 IF INKEY$="1" THEN GO TO 10
00
 190 IF INKEY$="2" THEN GO TO 20
00
 200 IF INKEY$="3" THEN GO TO 30
00
 210 IF INKEY$="4" THEN GO TO 40
00
 220 IF INKEY$="5" THEN GO TO 50
00
 230 GO TO 180
1000 REM criar
1003 LET CF=CF+1
1005 CLS
1080 PRINT AT 1,1;"FICHA DO DISC
O NUMERO-";CF
1090 PRINT AT 3,1;"NOME DO DISCO
-"
1094 INPUT "NOME DO DISCO-";C$(C
F)
1095 PRINT AT 3,15;C$(CF)
1100 PRINT AT 4,1;"ANO DE COMPRA
-"
1104 INPUT "ANO DA COMPRA-";E(CF
)
1105 PRINT AT 4,15;E(CF)
1110 PRINT "MUSICAS DO LADO A"
1120 INPUT "QUANTAS MUSICAS TEM
O LADO A?";A1
1123 IF A1>10 THEN GO TO 1120
1125 FOR B=1 TO A1
1130 INPUT "MUSICAS DO LADO A-";
M$(CF,B)
1135 PRINT B;"-";M$(CF,B)
1140 NEXT B
1150 PRINT "MUSICAS DO LADO B"
1160 INPUT "QUANTAS MUSICAS TEM
O LADO B?";B1
1170 IF B1>10 THEN GO TO 1160
1175 FOR B=1 TO B1
1180 INPUT "MUSICAS DO LADO B-";
N$(CF,B)
1185 PRINT ,B;"-";N$(CF,B)
1190 NEXT B
1195 INPUT "ESTA CERTO(S/N)?";Y$
1196 IF Y$="S" THEN GO TO 1200
1197 IF Y$="N" THEN GO TO 1005
1198 GO TO 1195
1200 INPUT "MAIS FICHAS(S/N)";A$
1210 IF A$="S" THEN GO TO 1000
1220 IF A$="N" THEN GO TO 100
1230 GO TO 1200
2000 REM consultar
2005 CLS
2010 PRINT "TODO O FICHEIRO(S/N)
"
2020 IF INKEY$="S" THEN GO TO 20
45
2030 IF INKEY$="N" THEN GO TO 25
00
2040 GO TO 2020
2045 CLS
2050 FOR I=1 TO CF
2060 PRINT "FICHA N.";I
2070 PRINT "NOME DO DISCO-";C$(I
)
2080 PRINT "DATA DE COMPRA(ANO)-
";E(I)
2090 PRINT "MUSICAS DO LADO A"
2092 PRINT
2095 FOR B=1 TO 10
2096 PRINT B;"-";M$(I,B)
2097 NEXT B
2100 PRINT "MUSICAS DO LADO B"
2105 PRINT
2110 FOR A=1 TO 10
2120 PRINT A;"-";N$(I,A)
2130 NEXT A
2131 PRINT "UMA TECLA PARA CONTI
NUAR": PAUSE 0: CLS
2132 NEXT I
2200 INPUT "OUTRA VEZ(S/N)";A$
2210 IF A$="S" THEN GO TO 2000
2220 IF A$="N" THEN GO TO 100
2230 GO TO 2200
2500 CLS
2510 INPUT "NOME DO DISCO ?";D$
2512 GO SUB 2800
2520 FOR I=1 TO CF
2530 IF C$(I)=D$ THEN GO TO 2600
2540 NEXT I
2550 PRINT "NAO EXISTE!"
2560 GO TO 2500
2600 LET N=I: GO SUB 9500
2630 GO TO 3110
2800 FOR I=LEN D$ TO 19
2810 LET D$=D$+" "
2820 NEXT I
2830 RETURN
3000 REM alterar
3005 CLS
3010 INPUT "NUMERO DA FICHA-";N
3020 CLS : PRINT "FICHA NUMERO-"
;N
3025 PRINT "1-NOME DO DISCO-";C$
(N)
3030 PRINT "2-ANO DE COMPRA-";E(
N)
3040 PRINT "3-MUSICAS DO LADO A"
3045 PRINT
3050 FOR B=1 TO 10
3060 PRINT B;"-";M$(N,B)
3070 NEXT B
```

```
3075 PRINT "4-MUSICAS DO LADO B-
"
3078 PRINT
3080 FOR B=1 TO 10
3090 PRINT B;"-";N$(N,B)
3100 NEXT B
3110 PRINT ''"0-PARA VOLTAR AO ME
NU": PRINT "QUAL O CAMPO A ALTER
AR(1,2,3,4,0)"
3120 INPUT Y$
3130 IF Y$="1" THEN INPUT "NOME
DO DISCO-";C$(N): GO TO 3020
3140 IF Y$="2" THEN INPUT "ANO D
E COMPRA-";E(N): GO TO 3020
3150 IF Y$="3" THEN GO TO 3200
3160 IF Y$="4" THEN GO TO 3300
3170 IF Y$="0" THEN GO TO 100
3200 INPUT "NUMERO DA MUSICA-";I
3210 INPUT "NOVO NOME PARA A MUS
ICA-";U$
3230 LET M$(CF,I)=U$
3240 GO TO 3020
3300 INPUT "NUMERO DA MUSICA-";I
3320 INPUT "NOVO NOME PARA A MUS
ICA-";U$
3330 LET N$(CF,I)=U$
3400 GO TO 3020
4000 REM gravar e verificar
4005 CLS
4010 PRINT "TEM TUDO PREPARADO?
(S)"
4020 INPUT Y$
4030 IF Y$="S" THEN GO TO 4070
4040 IF Y$="N" THEN GO TO 100
4050 GO TO 4020
4070 CLS : PRINT "G-GRAVAR": PRI
NT "R-REPRODUZIR"
4080 INPUT Y$
4090 IF Y$="G" THEN GO TO 4115
4100 IF Y$="R" THEN GO TO 4200
4110 GO TO 4080
4115 LET E(98)=CF
4120 SAVE "FIC." DATA C$(): SAVE
 "FIC.-1" DATA E(): SAVE "FIC.-2
" DATA M$(): SAVE "FIC.-3" DATA
N$()
4130 GO TO 100
4200 LOAD "" DATA C$(): LOAD ""
DATA E(): LOAD "" DATA M$(): LOA
D "" DATA N$()
4210 LET O=9000: LET CF=E(98): G
O TO 100
5000 REM imprimir
5005 CLS
5006 CLS : PRINT "CARREGAR EM I
PARA IMPRIMIR"
5007 IF INKEY$="I" THEN GO TO 50
10
5008 PAUSE 0: IF INKEY$<>"I" THE
N GO TO 5006
5010 FOR I=1 TO CF
5020 LPRINT "NOME DO DISCO-";C$(
I)
5030 LPRINT "DATA DE COMPRA(ANO)
-";E(I)
5040 LPRINT "MUSICAS DO LADO A"
5045 FOR B=1 TO 10
5046 LPRINT B;"-";M$(I,B)
5047 NEXT B
5050 LPRINT "MUSICAS DO LADO B"
5070 FOR A=1 TO 10
5080 LPRINT A;"-";N$(I,A)
5090 NEXT A
5100 LPRINT "=====================
=============="
5900 PRINT "UMA TECLA PARA VOLTA
R AO MENU": PAUSE 0: GO TO 100
9500 CLS
9510 PRINT "FICHA NUMERO-";N
9520 PRINT "NOME DO DISCO-";C$(N
)
9530 PRINT "ANO DE COMPRA-";E(N)
9535 PRINT "MUSICAS DO LADO A"
9540 FOR B=1 TO A1
9550 PRINT B;"-";M$(N,B)
9560 NEXT B
9565 PRINT "MUSICAS DO LADO B"
9570 FOR B=1 TO B1
9580 PRINT B;"-";N$(N,B)
9590 NEXT B
9600 PRINT "UMA TECLA PARA VOLTA
R AO MENU"
9610 PAUSE 0: GO TO 100
```

STARS WAR

SPECTRUM 16 K

Autor: HUGO ASSUNÇÃO
Lisboa

INSTRUÇÕES

```
 o objectivo deste jogo e

destruir as naves imperiais"
que se aproximam da sua base

 para mover a sua nave use

ESQUERDA-"5"----"8"-DIREITA

 para disparar use a tecla"0"

 so pode deixar passar 5 naves


PRIMA QUALQUER TECLA PARA JOGAR
```

```
  5 LET b=BIN 00100000: LET i=B
IN 00000100
 10 FOR n=144 TO 154
 20 FOR m=0 TO 7
 30 READ a: POKE USR CHR$ (n)+m
,a
 40 NEXT m: NEXT n
 50 DATA BIN 00100001,BIN 01110
001,BIN 00100001,BIN 00100001,BI
N 00100001,BIN 00100001,BIN 1010
0011,BIN 11100010
 60 DATA BIN 10000100,BIN 10001
110,BIN 10000100,BIN 10000100,BI
N 10000100,BIN 10000100,BIN 1100
0101,BIN 01000111
 70 DATA BIN 10100010,BIN 10110
010,BIN 10111010,BIN 10111111,BI
N 10011111,BIN 10001110,BIN 1111
1011,BIN 11000000
 80 DATA BIN 01000101,BIN 01001
101,BIN 01011101,BIN 11111101,BI
N 11111001,BIN 01110001,BIN 1101
1111,BIN 00000011
 90 DATA BIN 00110001,BIN 01100
111,BIN 01101111,BIN 11111100,BI
N 11111100,BIN 01101111,BIN 0110
0111,BIN 00110010
100 DATA BIN 10001100,BIN 11100
110,BIN 11110110,BIN 00111111,BI
N 00111111,BIN 11110110,BIN 1110
0110,BIN 01001100
```

```
 110 DATA BIN 0,BIN 01000010,BIN
 00001000,BIN 10011101,BIN 10011
101,BIN 00001000,BIN 01000010,0
 120 DATA BIN 00001001,BIN 01000
000,BIN 00111111,BIN 10110100,BI
N 11111100,BIN 00111110,BIN 0100
0010,BIN 00010001
 130 DATA BIN 01010101,BIN 00110
100,BIN 11111111,BIN 01111110,BI
N 11101001,BIN 01111110,BIN 0010
1001,BIN 00100101
 150 DATA BIN 0,b,BIN 01110000,b
,b,b,b,0
 170 DATA BIN 0,c,BIN 00001110,c
,c,c,c,0
 180 LET hs=0
 190 LET z=0
 200 LET a=20: LET b=11
 220 LET s=0
 230 LET c=1: LET d=15+INT (RND*
15)-INT (RND*15)
 240 BORDER 0: PAPER 0: CLS
 260 PRINT INK 6; PAPER 0;AT 0,1
0;"INSTRUCOES"
 270 PRINT PAPER 0; BRIGHT 1; IN
K 7;"   O objectivo deste jogo
e   destruir as naves imperi
ais";: INK 6;"";: INK 7;"  q
ue se aproximam da sua base !  
  Para mover a sua nave use  "
 280 PRINT INK 5;TAB 14;"";: I
NK 7;"ESQUERDA-";: FLASH 1;"5";:
 FLASH 0;"--";: INK 5;"";: INK
7;"--";: FLASH 1;"8";: FLASH 0;"-
-DIREITA"
 290 PRINT INK 7;"   para dispar
ar use a tecla";: FLASH 1;"0";: F
LASH 0;"  ";: INK 5; PAPER 0: FLAS
H 1;"  so pode deixar passar 5
naves "
 300 PRINT FLASH 1; BRIGHT 0; IN
K 4;AT 21,0;"PRIMA QUALQUER TECL
A PARA JOGAR": FOR n=0 TO 12 STE
P 2: FOR f=0 TO 12 STEP 2: FOR g
=0 TO 12 STEP 2
 305 BEEP .1,f-g+n: BEEP .1,f+g-
n: BEEP .1,n-f+g
 306 IF INKEY$<>"" THEN GO TO 31
0
 307 NEXT g: NEXT f: NEXT n
 310 BORDER 0: PAPER 0: CLS
 500 PRINT INK 6;AT 0,14;"": P
RINT FLASH 1; INK 2; BRIGHT 1;AT
 0,12;5-z
 505 PRINT BRIGHT 1; INK 4;AT 0,
0;"SCORE  ";: FLASH 1;s;: FLASH 0;
 INK 6; BRIGHT 1;AT 0,19;"HI SCO
RE  ";: FLASH 1;hs
 510 LET b=b+(INKEY$="8" AND b<=
27)-(INKEY$="5" AND b>0)
 520 PRINT BRIGHT 1; INK 5;AT a,
b;"";: BRIGHT 1; INK 5;AT a+1
,b;""
 525 PRINT INK 0; PAPER 0;AT c,d
;""
 530 LET c=c+1: LET d=d-INT (RND
*4)-INT (RND*4)
 535 IF d<0 THEN LET d=0
 536 IF d>30 THEN LET d=30
 540 PRINT INK 6; BRIGHT 1; FLAS
H 0;AT c,d;""
 555 IF INKEY$="5" OR INKEY$="8"
 THEN BEEP .01,10
 560 IF INKEY$="0" THEN GO SUB 8
000
 570 IF d<0 THEN LET d=0: IF d>3
0 THEN LET d=30
 580 IF c>=20 AND d>=b AND d<=b+
2 THEN GO TO 2000
 590 IF c=21 THEN GO TO 1000
 600 GO TO 500
1000 LET z=z+1: BEEP .2,10: BEEP
 .2,0
1005 IF z=5 THEN GO TO 3000
1010 PRINT AT c,d; INK 0; PAPER
0;" "
1020 LET c=1: LET d=15+INT (RND*
14)-INT (RND*14)
1030 GO TO 540
2000 FOR n=0 TO 10
2010 PAPER 5: CLS : PAUSE 3
2020 PAPER 2: CLS : PAUSE 3
2030 NEXT n
2040 PAPER 0: BRIGHT 1: CLS
2050 PRINT INK 4;AT 0,0;"SCORE  
";: FLASH 1;s
2060 PRINT INK 6;AT 0,19;"HI SCO
RE  ";: FLASH 1;hs
2070 IF s<=hs THEN GO TO 3000
2075 FOR n=0 TO 7: BORDER n: BEE
P .7,10+n*2: NEXT n
2080 FOR n=hs TO s STEP 1: BEEP
.005,35: PRINT FLASH 1; INK 6;AT
 0,29;n: NEXT n
2090 LET hs=n-1
2100 PRINT INK 4; FLASH 1;AT 5,4
;"Bateu o recorde de hoje"
3000 PRINT INK 2; FLASH 1;AT 10,
10;"FIM DO JOGO"
3010 PAUSE 500: GO TO 180
3000 CLS : PRINT INK 5; FLASH 1;
AT 10,0;" DEIXOU PASSAR MAIS DE
5  NAVES                INIMIGAS
   "
3010 GO TO 2050
8000 FOR x=19 TO 2 STEP -1
8005 BEEP .005,x*2+10
8010 PRINT BRIGHT 1; INK 7; FLAS
H 0;AT x,b+1;"↑↑": PRINT BRIGHT
0; INK 0; PAPER 0;AT x,b+1;"  "
8020 IF x=c AND (d=b+1 OR d=b+2
OR d=b) THEN GO TO 8500
8030 NEXT x
8035 GO SUB 9700
8040 RETURN
8505 IF d=b+1 THEN GO TO 8600
8510 LET s=s+1: GO SUB 9700
8515 PRINT INK 0; PAPER 0;AT c,d
;"  ": LET c=1: LET d=15+INT (RN
D*14)-INT (RND*14)
8520 GO TO 8040
8600 LET s=s+3: GO SUB 9700
8605 PRINT INK 0; PAPER 0;AT c,d
;"  ": LET c=1: LET d=15+INT (RN
D*14)-INT (RND*14)
8610 GO TO 8040
9700 PRINT INK 6;AT x,b+1;"":
BEEP .15,15: PRINT INK 6; BRIGHT
 1;AT x,b+1;"": BEEP .15,10: P
RINT INK 2; BRIGHT 1;AT x,b+1;"
": BEEP .15,5
9710 PRINT INK 0; PAPER 0;AT x,b
+1;"  "
9720 RETURN
```

NOVO LIVRO

LA CONDUITE DU ZX SPECTRUM

HARTNELL Tim e JONES Dilwyn

Eyrolles, Paris, 1983

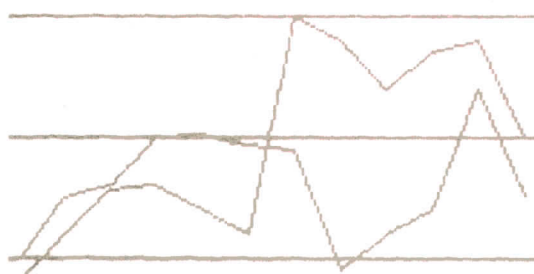
— O teclado; Instruções (PRINT, TAB, SAVE, VERIFY, MERGE . . .); Variáveis; Cor e Gráficos; N.os Cadeias de caracteres; PEEK e POKE . . .

Preço (Fotocópias) 860$00

Desconto de 10% para Sócios do Clube

TRAÇADO DE GRÁFICOS

SPECTRUM

```
 10 LET s=PI↑50
 15 DIM y(12)
 20 INPUT  INVERSE 1;"Nome do grafico "; INVERSE 0;"(32 caracteres) "; LINE t$:
IF LEN t$>32 THEN  GO TO 10
 30 PRINT AT 0,0;t$;: FOR m=LEN t$ TO 31: PRINT " ";: NEXT m
 40 FOR n=0 TO 22: PRINT AT 20,n;" ";: NEXT n: PRINT
 50 LET a$="JFMAMJJASOND"
 60 FOR n=1 TO 12: PRINT a$(n);" ";: NEXT n
 70 PRINT AT 2,24; INVERSE 1;"Entrada";AT 3,24; INVERSE 0;"max.150"
 80 FOR n=1 TO 12: PRINT AT n+4,24;a$(n);" = ";y(n);"    ": NEXT n
100 FOR n=1 TO 12
110 INPUT  INVERSE 1;"Entradas"; INVERSE 0;" (s=fim) ";y(n): IF y(n)=s THEN  GO
TO 170
120 IF y(n)>150 OR y(n)<0 THEN  PRINT AT 18,24; FLASH 1;" Fora ";AT 19,24;"  do
 ";AT 20,24;"limite": PAUSE 100: FOR m=18 TO 20: PRINT AT m,24; FLASH 0;"
": NEXT m: GO TO 110
130 PRINT AT n+4,24; INVERSE 1;a$(n); INVERSE 0;AT n+4,28;y(n)
140 IF n=1 THEN  PLOT 4,y(1)+15
150 IF n>1 THEN  DRAW 16*(n-1)-16*(n-2),y(n)-y(n-1)
160 NEXT n
170 INPUT "Tracado da escala ? "; LINE g$: IF g$<>"s" THEN  GO TO 200
180 FOR y=0 TO 100 STEP 50: PLOT 0,65+y: DRAW 184,0: NEXT y
200 INPUT "Copiar na impressora ? "; LINE p$
210 IF p$="s" THEN  COPY
300 INPUT "novos dados ? "; LINE r$
310 IF r$<>"s" THEN  STOP
320 INPUT "Grafico sobreposto ? "; LINE s$
330 IF s$="s" THEN  GO TO 10
340 CLS : GO TO 10
```

TRAÇADO DE ESCALAS (ANUAL)

Este programa elabora escalas do tipo comercial com a vantagem de, com maior facilidade, verificar a variação de valores.

Assim terá apenas de introduzir os valores correspondentes aos meses do ano representados pelas iniciais. Em seguida, ser-lhe-á dada a opção do traçado na escala e, se deseja, cópia do gráfico.

Por fim, poderá dar entrada de novos dados, conseguindo obter a sobreposição dos gráficos caso o pretenda.

CARACTERES DE DUPLA ALTURA

SPECTRUM

Adapt. ALEXANDRE SOUSA
Porto

Descritivo: O método que o autor (Kevin Ball's) escolheu, está baseado no estabelecimento de uma rotina, que examina a área do 'Display File' e expande os bytes de modo a produzir caracteres de dupla altura.

OBSERVAÇÃO: TAMBÉM POSSUÍMOS IDÊNTICO PROGRAMA PARA O ZX 81/TS 1 000, E PODEMOS ENVIAR UMA CÓPIA AOS ASSINANTES QUE O PEDIREM.

MÉTODO: Inicialmente escrevemos o caracter ou o texto no ecran. Indicamos a posição da linha e a coluna onde queremos o caracter (ou o primeiro caracter do texto). Em seguida, a rotina é chamada na linha 1 200... RANDOMIZE USR 32 500.

Observe que cada caracter ocupa duas linhas.

O programa está projectado de forma a que o utilizador não se preocupe com a posição da Memória RAM em que vai alojar a rotina, quer tenha 48 K ou 16 K.

SE PEDIR PARA ESCREVER NA LINHA 21, OS CARACTERES NÃO PODEM SER EXPANDIDOS POR FALTA DE ESPAÇO NO ECRAN.

COMO O CÓDIGO DECIMAL ESTÁ GUARDADO NAS LINHAS DE 'DATA', PODERÁ USAR ESTE PROGRAMA DENTRO DOS PRÓPRIOS PROGRAMAS, NECESSITANDO APENAS DE RENUMERAR AS LINHAS E CHAMAR ESTA ROTINA.

```
ESTE PROGRAMA PERMITE ESCREVER
QUALQUER CARACTER OU FRASE, MAS
COM DUPLA ALTURA

Por exemplo : EU SOU O MAIOR !

EU SOU O MAIOR !
```

```
   1 REM PROGRAMA P/CARACTERES
DE DUPLA ALTURA
  10 CLEAR 32499
  20 LET A=32500
  30 READ Z: IF Z<>999 THEN POKE
 A,Z: LET A=A+1: GO TO 30
  40 DATA 237,75,176,92,121,254,
32,56
  50 DATA 1,201,120,254,21,56,1,
201
  60 DATA 205,158,14,197,6,0,9,1
93
  70 DATA 229,229,120,80,205,158
,14,8
  80 DATA 0,9,209,1,0,7,9,235
  90 DATA 8,229,6,4,126,18,21,18
 100 DATA 21,37,16,248,209,5,4,1
25
 110 DATA 18,21,18,21,37,16,248,
225
 120 DATA 124,15,15,15,230,3,246
,88
 130 DATA 103,126,1,32,0,9,119,2
01
 140 DATA 999
 142 CLS : PRINT "F)rase ou text
o": PRINT "C)aracter a caracter"
 144 PAUSE 0: IF INKEY$="F" THEN
 GO TO 500
 146 IF INKEY$="C" THEN GO TO 15
0
 148 GO TO 144
 150 INPUT "Caracter..? ";a$
 160 INPUT "Linha..? ";l
 170 INPUT "Coluna..? ";c
 180 GO SUB 1180
 210 GO TO 150
 500 CLS : PRINT "texto...? "
 510 INPUT t$
 520 INPUT "LINHA INICIAL=";L
 530 INPUT "COLUNA INICIAL=";C
 540 CLS : FOR I=1 TO LEN (T$)
 545 GO SUB 1000
 550 NEXT I
 600 PAUSE 0: GO TO 142
1000 IF C>=31 THEN LET L=L+2
1010 IF C>=31 THEN LET C=0
1020 LET A$=T$(I): LET C=C+1
1180 POKE 23728,L: POKE 23729,C
1190 PRINT AT L,C; FLASH 1;a$
1200 RANDOMIZE USR 32500
1210 RETURN
```

FIRE FOX (CORRECÇÃO)

(V. n.° 12, pág. 15)

A listagem publicada no n.° 12 está incorrecta. Aqui vai a listagem devidamente corrigida.

```
   1 PAPER 0: INK 7: CLS : BORDE
R 5
   2 CLEAR 65099: LET ss=0: LOAD
 ""CODE 65100
   3 GO TO 2000
   4 LET s=0
   5 LET fy=0: LET fx=0
   6 LET b=0: LET by=0: LET bx=0
  10 FOR n=16 TO 21: PRINT AT n,
0; PAPER 5; INK 5;"
                  ": NEXT n
  20 LET f=0: LET x=10: LET y=7
  30 PRINT AT 21,11; INK 0; PAPE
R 5; INVERSE 1;"FIREFOX";AT 19,1
1; INVERSE 0;"score ";s;AT 17,11
; FLASH 1;"Hi Score "; FLASH 0;
INVERSE 1;ss
 100 FOR z=1 TO 289
```

```
 120 READ n
 130 POKE 65190,n
 135 LET s=s+y
 136 PRINT PAPER 5; INK 1;AT 19,
17;s
 140 LET l=USR 65100
 160 LET y=y+(IN 49150=253)-(IN
57342=254)
 165 LET x=x+(IN 65278=251)-(IN
65278=253)
 167 IF x<0 THEN LET x=0
 168 IF y<0 THEN LET y=0
 169 IF ATTR (y,x)<>7 OR ATTR (y
,x+1)<>7 THEN LET k=1: FOR n=1 T
O 20: LET k=-k: BEEP .15,n+k: NE
XT n: GO TO 3000
 170 PRINT AT y,x; INK 7; BRIGHT
 1;"DO"
 172 IF n=7 THEN PRINT INK 4;AT
16-n,31;"H"
 173 FOR d=1 TO 2: NEXT d
 174 PRINT AT y,x;"  "
 175 NEXT z
 180 PRINT AT 6,8;"MISSAO COMPLE
TA": GO TO 3000
 310 RESTORE : GO TO 100
1000 DATA 1,1,1,1,1,1,1,1,1,1,1,
1,1,1,1,1,1,1,1,2,3,4,5,6,7,8,9,
10,11,12,13,14,13,12,11,10,9,9,9
,9,9,9,9,9
1010 DATA 9,9,9,9,9,9,9,9,10,10,
11,11,10,9,8,7,6,5,3,2,2,3,4,3,1
,2,3,4,5,6,7,8
1020 DATA 7,6,5,4,3,2,1,1,1,1,1,
1,2,2,2,2,2,3,4,4,3,3,3,3,3,4,4,
5,5,5,5,5,5,5,5,5,5,5,5,5,6,4,6,
6,6,6,6,7,7,7,7,7
1030 DATA 9,9,9,9,9,9,9,10,11,12
,13,14,15,15,15,15,14,14,13,13,1
2,12,11,11,11,11,9,7,5,3,4,5,6,7
1040 DATA 9,11,13,15,16,16,16,15
,14,14,14,14,14,14,14,14,16,16,1
4,12,11,9,7,5,3,1,1,3,3,5,6,8,9,
11
1050 DATA 8,6,4,2,2,1,1,1,1,1,1,
1,1,3,9,12,14,16,16,16,16,16,15,
15,15,15,15,15,16,15,15,15,15,13
,13,13,13,13,15,14,13,12,10,9,8,
7,6,5,4,3,3,2,1,1,1,1,1,3,3,4,2,
2,1
1070 DATA 1,1,1,1,1,1,1,1,1,1,1,
1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,
1,1,1
2000 RESTORE 2000
2010 DATA BIN 00001000,BIN 00001
000,BIN 0001100,BIN 00111110,BIN
 00111110,BIN 01111111,BIN 01101
011,BIN 00001000: FOR n=0 TO 7:
READ r: POKE USR "H"+n,r: NEXT n
2015 DATA BIN 0,BIN 00000,BIN 11
110000,BIN 11111000,BIN 1111111
,BIN 00011111,BIN 00000111,BIN 0
0001111: FOR n=0 TO 7: READ r: P
OKE USR "D"+n,r: NEXT n
2016 DATA BIN 0,BIN 0000,BIN 00,
BIN 0,BIN 11110000,BIN 11111100,
BIN 1111111,BIN 0000: FOR n=0 T
O 7: READ r: POKE USR "O"+n,r: N
EXT n
2020 RESTORE
2025 CLS
2040 INPUT "Esta pronto? (S)";s$
2050 GO TO 4
3000 RESTORE : INPUT "Outra tent
ativa? (S/N)";s$
3010 IF s>ss THEN LET ss=s
3015 CLS
3020 PRINT AT 17,20;ss
3050 IF s$(1)="n" OR s$(1)="N" T
HEN STOP
3060 GO TO 4
```

VU-METER

VU-METER (correcções)

(V. n.º 14, pág. 17)

Devido ao facto de a listagem ter sido efectuada após o "RUN" do programa, e como não podia funcionar com impressora ligada, essa listagem foi publicada com erros. Apresentamos agora a listagem correcta.

```
   1 REM !8\?7w COPY FLASH IF RE
TURN COPY !74
   2 INK 0: PAPER 7: CLS
   3 IF PEEK 23760<>33 THEN GO S
UB 8000
   4 LET b$="
 "
   5 PRINT INK 7; PAPER 1;AT 0,0
;"   VU METER   "
   6 PLOT 0,30: DRAW 200,0: DRAW
 0,-30: DRAW 47,0: DRAW 0,30: DR
AW -47,0: PLOT 0,30: DRAW 0,-30:
 DRAW 200,0
   7 PRINT AT 19,1;"-20   -10   0
  10   20"
   8 PRINT AT 19,26; INK 0; PAPE
R 4;"PEAK"
   9 LET A$="████████████████████
█"
  10 FOR N=0 TO 255
  20 LET l=USR 23760
  21 LET A=PEEK 23808
  29 INK 1
  30 PLOT N,87
  40 DRAW 0,(A/4)
  51 LET T=A/13
  52 INK 0
  60 PRINT INK 5;AT 20,1;A$( TO
T)+B$( TO (20-T))
  61 PRINT AT 20,12; OVER 1; PAP
ER 7; INK 2
  80 IF T>14 THEN PRINT AT 20,26
; INK 3; PAPER 7;"▬▬▬▬"
  81 FOR F=0 TO 3: NEXT F
  90 PRINT AT 20,26;"    "
 200 NEXT N: RUN
8000 RESTORE
8600 FOR F=23760 TO 23784
8605 READ A: POKE F,A: NEXT F
8800 DATA 33,56,12,62,0,119,6,23
8,219,250,254,255,40,1,52,16,247
,201,0,0,0,0,0,0,0
8810 RETURN
```

JOGO DO TRAÇO

NEW BRAIN

Adapt. do BEGINNER'S GUIDE
Por **Isabel Cristina**
PORTO

```
10  REM    ***Jogo do traco***
20  OPEN#0,0,"85": PUT 23,65: CLOSE#1
30  OPEN#1,11,"n130"
40  CLEAR: x3=168
50  plot colour(1)
60  plot range(200,100),centre(12,30)
70  plot background(1),wipe
80  PUT 31: REM limpar o ecran
90  PRINT"Jogo do TRACO "
100 PRINT: PRINT"Vamos jogar por turnos";   "- Tu comecas"
110 PRINT"De cada vez so podes escolher"
120 PRINT"1 , 2 ou 3 tracos do quadro"
130 PRINT"O que ficar com o ultimo  "
140 PRINT"Perde."
150 GOSUB 1000: x2=0
160 PRINT:PRINT"Quantos tracos escolhes ?"
170 PRINT"Introduz o numero e acciona NEW LINE "
180 INPUT (""")s$
190 PUT 31
200 IF s$="1" THEN x3=x2
210 IF s$="2" THEN x3=x2+14
220 IF s$="3" THEN x3=x2+28
230 IF s$<"1" OR s$>"3" THEN GOTO 160
240 GOSUB 1200
250 plot colour(2): GOSUB 1000
256 REM
257 RANDOMIZE
258 R = INT(RND * 4)
259 IF R = 0 THEN GOTO 258
260 IF x2 = 126 THEN GOTO 430
261 IF x2 = 140 THEN GOTO 440
262 IF x2 = 154 THEN GOTO 450
263 IF x2 = 168 THEN GOTO 630
264 IF x2 >= 182 THEN GOTO 680
270   IF R = 1 THEN GOTO 300
280   IF R = 2 THEN GOTO 310
290   IF R = 3 THEN GOTO 320
300   x3=x2+28: a=3: GOTO 330
310   x3=x2+14: a=2: GOTO 330
320   x3=x2:a=1
330   PRINT"E a minha vez ! ":PRINT"Escolho  ";
340   GOSUB 1200: GOSUB 1000
350   PUT 31
360   PRINT"Joga outra vez !": PRINT
370   PRINT"Quantos tracos escolhes"
380   PRINT"Introduz o numero e acciona NEW LINE"
390   GOTO 180
410 IF s$="2" THEN GOTO 440
420 IF s$="3" THEN GOTO 450
430 x3=x2+28: a=3: GOTO 460
440 x3=x2+14: a=2: GOTO 460
450 x3=x2: a=1
460 PUT 31
470 PRINT"Sou eu ! - e escolho ";a
480 GOSUB 1200: GOSUB 1000: PRINT: PRINT
490 PRINT "Ficou o ultimo para ti"
500 PRINT: PRINT"GANHEI!",,,,,,,,,
510 PRINT"Queres jogar outra vez ?";"(S,N)"
530 INPUT (""")s$
540 IF s$="s" OR s$="S" THEN GOTO 40
550 IF s$="n" OR s$="N" THEN GOTO 580
560 GOTO 530
580 PUT 31: plot wipe: PRINT
590 PRINT "ADEUS  !"
600 CLOSE#1
610 END
630 PUT 31: PRINT: PRINT "Fica o ultimo"
640 PRINT "Para mim .": PRINT
650 PRINT "OK! - GANHASTE"
660 PRINT: GOTO 510
680 PUT 31: PRINT
690 PRINT "Escolheste muitos - ";
700 PRINT: PRINT "GANHEI  ."
710 PRINT: GOTO 510
1000 REM
1005 FOR x = x2 TO x3 STEP 14
1010   plot place(x,0)
1020   plot move(x,50)
1030   plot move(x+4,50)
1040   plot move(x+4,0)
1050   plot move(x,0)
1060   FOR x1=46 TO 49
1070   plot place(x,x1),move(x+4,x1)
1080   NEXT x1
1090 NEXT x
1100 x2 = x3 + 14
1110 RETURN
1200 REM delay loop
1210 FOR z = 1 TO 600
1220 NEXT z
1230 RETURN
```

ESPAÇO SPECTRUM

Autor: FERNANDO D'ALMEIDA PRECES

(Parte III)

CAPÍTULO 1.º — PROGRAMA 3

Actualmente no mercado, existem centenas de programas monitores quer em revistas de várias nacionalidades, quer em cassetes gravadas; alguns deles bem sofisticados, outros mais simples, mas quase todos programados em linguagem máquina.

Ao ser criado no ESPAÇO SPECTRUM um capítulo destinado a programas monitores, pretendeu-se dar ao nosso leitor alguns auxiliares eficazes para a elaboração dos *seus programas,* sem contudo esquecer o problema capital que consiste na falta de conhecimentos técnicos da generalidade dos utilizadores portugueses. Há portanto necessidade que cada programa monitor não seja somente uma ferramenta de trabalho mas também uma fonte de conhecimentos, possuindo uma estrutura de programação acessível que possa mais tarde ser alterada ao gosto de cada um. Para tal tem sido seleccionados programas cuja estrutura seja o BASIC, linguagem que ainda hoje é considerada como a mais próxima ao homem.

Este programa não foge a essa regra, apesar de possuir uma pequena rotina máquina. A parte do programa que se destina ao desenho e pintura do ecran está toda programada em BASIC. A rotina em C. M. comporta apenas as instruções necessárias para elaborar letras de formato (n) com tamanho e posição à escolha do utilizador. Contém esta algumas instruções algo complexas, havendo por isso necessidade de adiar a descrição do seu funcionamento para mais tarde.

O programa possui todas as instruções necessárias para a execução do trabalho a que se propõe, mas mais adiante irá sofrer pequenas modificações, e algumas das rotinas em BASIC irão passar a código máquina o que o tornará um programa muito rápido e completo.

POR QUESTÕES DE ESPAÇO, A PARTIR DO 2.º CAPÍTULO OS TEXTOS INTRODUÇÃO À LINGUAGEM MÁQUINA E ESPAÇO SPECTRUM PASSARÃO A FORMAR UM ÚNICO ARTIGO — INTRODUÇÃO À LINGUAGEM MÁQUINA ZX81 E SPECTRUM.

Noção teórica sobre o ecran do SPECTRUM em alta definição (ou resolução)

Na sua execução normalizada o Spectrum divide o ecran em duas partes distintas:

a) Ecran Superior (composto de 22 linhas de 32 colunas)
b) Ecran Inferior (composto de 2 linhas de 32 colunas)

A intercepção de duas linhas com duas colunas define o formato do espaço reservado aos caracteres, ou seja, em 22 linhas com 32 colunas existe um espaço que pode ser preenchido por 704 caracteres. Neste computador, o espaço preenchido por um caracter está subdividido em 64 fracções iguais, ou seja 8 * 8 espaços iguais. Cada um destes espaços é chamado um "pixel".

Pelo manual do Spectrum já sabemos também que é possível, com a instrucção PLOT, fazer aparecer no ecran um único pixel. Claro que, como veremos, há outros processos para executar o mesmo trabalho.

Também pelo manual sabemos que o Spectrum pode imprimir 256 pixeis na horizontal por 176 na vertical e que essa malha está dentro dos limites definidos por alta resolução.

Para reter na memória cada um desses pixeis poder-se-á pensar, à primeira vista, que são necessários 256 * 176 = 45056 endereços diferentes. Na realidade, apenas são precisos 6144.

Vamos ver porquê? Claro que muitos já conhecem a razão. Mas vamos esclarecer os outros.

Um endereço no Spectrum armazena um byte (8 bits) e por isso pode albergar qualquer número compreendido entre 0 e 255.

Esses 8 bits podem, como todos sabemos, tomar valores que representam as 256 combinações binárias possíveis. O mesmo é dizer que as combinações possíveis de 8 pixeis podem ser armazenadas num único endereço. Então o espaço necessário na memória será

(256/8) = 32 localizações
32 grupos * 176 pixeis verticais = 5632 localizações para o ecran superior

(256/8) = 32
32 grupos * 16 pixeis verticais = 512 localizações para o ecran inferior
5632 + 512 = 6144 localizações

No Spectrum, o ficheiro de projecção é guardado na RAM nos endereços (16384 a 22528) e nos endereços (22529 a 23296), espaço reservado aos atributos. Nestes não vamos ainda falar.

O programa imprime pixeis no ecran com a cor da tinta pretendida, desenha círculos, quadrados, rectângulos, linhas, etc.

Quando o ecran estiver a seu gosto, pode gravá-lo em (bytes SCREEN$) numa cassete, bastando para isso premir a tecla adequada.

```
  10 LET z=0: LET x=127: LET y=8
7: LET a=127: LET b=87: LET cf=z
: LET j=1
  14 GO SUB 810: CLS
  20 INPUT "Indique a cor para o
 papel(0-7) "; LINE i$: GO SUB 1
600: IF i<0 OR i>7 THEN GO TO 20
  24 LET pc=i
  30 INPUT "Indique a cor para a
 tinta(0-7) "; LINE i$: GO SUB 1
600: IF i<0 OR i>7 THEN GO TO 30
  34 LET ic=i: LET ik=i
  40 INK ic: BORDER pc: PAPER pc
: CLS
  50 DIM r$(1000): LET s=1
  60 LET s$="qijclbCB"
  64 LET t$="234567"
  65 PRINT AT 10,0;"Quer desenho
 ou letras? (d ou l)": INPUT l$
  66 IF l$="l" THEN GO TO 8600
  67 CLS
  70 GO SUB 3480
  80 GO SUB 2070
 100 PAUSE z: LET i$=INKEY$
 110 FOR n=1 TO 6: IF n>2 AND s>
990 THEN LET p$="          NAO HA M
AIS ESPACO          ": GO SUB 3420:
 GO SUB 3480: GO TO 100
 114 IF s$(n)=i$ THEN GO TO 100+
VAL t$(n)
 130 NEXT n
 140 LET i=CODE i$: IF i>7 AND i
<12 THEN GO SUB 2070: GO TO 240+
10*(i-8)
 150 IF i>52 AND i<57 THEN GO SU
B 2070: GO TO 340+10*(i-53)
 160 GO TO 100
 200 LET xs=2: LET ys=2: LET p$=
"   TERMINADO    ": LET yy=175: G
O SUB 3500: PAUSE 100: INPUT "QU
ER GRAVAR O ECRAN? (s ou n)"; l$
 210 IF l$="s" THEN GO SUB 2070:
 GO TO 8500
 220 STOP
 240 LET x=x-j: IF x<4 THEN LET
x=4
 244 GO TO 80
 250 LET x=x+j: IF x>251 THEN LE
T x=251
 254 GO TO 80
 260 LET y=y-j: IF y<4 THEN LET
y=4
 264 GO TO 80
 270 LET y=y+j: IF y>171 THEN LE
T y=171
 274 GO TO 80
```

```
55: LET w=USR 32256: RETURN
3600 LET p$="
         ": LET ys=2: LET yy
=176: GO TO 3500
8000 CLEAR 32255: LOAD "c"CODE
8010 BORDER 5: PAPER 5: INK 1: C
LS : LET xs=4: LET ys=4: LET yy=
20: LET p$="DESENHO": GO SUB 350
0
8020 INK 2: LET p$="COLORIDO": L
ET xs=2: LET ys=3: LET yy=70: GO
 SUB 3500: FLASH 0
8030 INK 3: LET yy=120: LET xs=3
: LET ys=4: LET p$="A ARTE NO EC
RAN": GO SUB 3500
8040 INK 0: LET yy=170: LET xs=1
: LET ys=2: LET p$="Prima uma te
cla": GO SUB 3500
8050 PAUSE 0: RUN
8500 SAVE "ecran"SCREEN$ : STOP
8600 BORDER 5: PAPER 5: CLS
8610 INPUT "Cor: ";t: INPUT "Lar
gura: ";xs: INPUT "Altura: ";ys:
 INPUT "Posicao: ";yy: INPUT "Le
tras: ";p$: INK t: GO SUB 3500
8620 GO TO 8610
9990 CLEAR : SAVE "desenho" LINE
 8000: SAVE "c"CODE 32256,300
```

```
CODIGO MAQUINA - 300 BYTES -INIC
IO DO CODIGO...32256
```

```
ANTES DE INTRODUZIR O CODIGO MAQ
UINA NAO SE ESQUECA DE FAZER ...
CLEAR 32255
```

```
32256=33   32257=15   32258=91
32259=126  32260=35   32261=34
32262=0    32263=91   32264=111
32265=50   32266=200  32267=33
32268=0    32269=41   32270=41
32271=41   32272=237  32273=75
32274=54   32275=92   32276=9
32277=62   32278=8    32279=50
32280=4    32281=91   32282=50
32283=11   32284=91   32285=50
32286=8    32287=91   32288=50
32289=10   32290=91   32291=50
32292=8    32293=91   32294=62
32295=8    32296=50   32297=5
32298=91   32299=126  32300=35
32301=34   32302=2    32303=91
32304=7    32305=50   32306=6
32307=91   32308=58   32309=5
32310=91   32311=61   32312=32
32313=50   32314=58   32315=4
32316=91   32317=61   32318=32
32319=24   32320=58   32321=14
32322=91   32323=71   32324=58
32325=12   32326=91   32327=73
32328=58   32329=10   32330=91
32331=122  32332=5    32333=32
32334=252  32335=50   32336=10
32337=91   32338=42   32339=0
32340=91   32341=105  32342=3
32343=126  32344=50   32345=4
32346=91   32347=58   32348=13
32349=91   32350=71   32351=58
32352=8    32353=91   32354=126
32355=50   32356=3    32357=91
32358=42   32359=2    32360=91
32361=105  32362=32   32363=126
32364=50   32365=5    32366=91
32367=58   32368=12   32369=91
32370=71   32371=58   32372=9
32373=91   32374=50   32375=7
32376=91   32377=58   32378=13
32379=91   32380=79   32381=197
32382=205  32383=164  32384=126
32385=193  32386=58   32387=7
32388=91   32389=60   32390=50
32391=7    32392=91   32393=13
32394=32   32395=241  32396=58
32397=8    32398=91   32399=60
32400=50   32401=8    32402=91
32403=5    32404=32   32405=221
32406=58   32407=6    32408=91
32409=195  32410=46   32411=126
32412=128  32413=64   32414=32
32415=16   32416=8    32417=4
32418=2    32419=1    32420=58
32421=142  32422=92   32423=238
32424=255  32425=71   32426=58
32427=141  32428=92   32429=160
32430=71   32431=58   32432=3
32433=91   32434=230  32435=248
32436=111  32437=58   32438=7
32439=91   32440=254  32441=192
32442=208  32443=31   32444=31
32445=31   32446=230  32447=31
32448=103  32449=203  32450=28
32451=203  32452=29   32453=203
32454=28   32455=203  32456=29
32457=203  32458=28   32459=203
32460=29   32461=62   32462=88
32463=130  32464=103  32465=58
32466=142  32467=92   32468=166
32469=178  32470=119  32471=58
32472=7    32473=91   32474=71
32475=230  32476=7    32477=246
32478=64   32479=103  32480=120
32481=31   32482=31   32483=31
32484=230  32485=24   32486=180
32487=103  32488=120  32489=23
32490=23   32491=230  32492=224
32493=111  32494=58   32495=3
32496=91   32497=71   32498=31
32499=31   32500=31   32501=230
32502=31   32503=181  32504=111
32505=235  32506=33   32507=156
32508=125  32509=120  32510=230
32511=7    32512=79   32513=6
32514=0    32515=9    32516=70
32517=26   32518=33   32519=6
32520=91   32521=203  32522=70
32523=40   32524=3    32525=175
32526=16   32527=201  32528=47
32529=176  32530=47   32531=16
32532=201  32533=0    32534=0
32535=0    32536=0    32537=0
32538=0    32539=0    32540=0
32541=0    32542=0    32543=0
32544=0    32545=0    32546=0
32547=23   32548=220  32549=10
32550=205  32551=11   32552=231
32553=80   32554=26   32555=23
32556=0    32557=0    32558=0
```

(Cont. no próximo número)

SIMULAÇÃO DE CIRCUITOS LÓGICOS

SPECTRUM/ZX81/TS 1000

Autor: MALCOM FARNSWORTH
Adapt. de **Alexandre Sousa**

Parte I

PROGRAMA: SIMULAÇÃO DE CIRCUITOS LÓGICOS

DESCRITIVO: Este programa é interessante para todos os que gostam de Electrónica Digital, de Lógica ou de Matemática (conhecendo os dispositivos usados na Electrónica . . .)

Sob o ponto de vista educacional, pensamos que o programa pode ser melhorado e articulado, de molde a incluir testes para alunos ou substituir situações reais (simulação), de modo a observar o tipo de resposta a determinadas combinações lógicas dos dispositivos:

AND—OR—NAND—NOR—INVERTER—OR EX—FLIP/FLOP

ESTE PROGRAMA FOI ESCRITO PARA O ZX 81 MAS É FACILMENTE ADAPTADO AO SPECTRUM

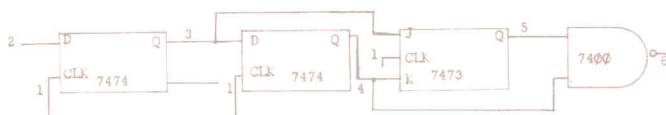
NOTA: As linhas 7510 a 7750 — são exemplificativas da possibilidade de desenhar o próprio esquema; no entanto, sob o ponto de vista prático, parece-nos de difícil generalização, pelo que pode ser omitida a linha 4030 e a rotina apontada (7510 . . . 7750). As linhas 5130, 6100, 7130 podem ser substituidas por uma rotina do tipo:

```
8000 input y$
8010 cls
8020 goto 4010
```

impedindo, desta forma, que o programa fique com STOP.

UTILIZAÇÃO:

Cada dispositivo lógico (AND — OR — NAND — FLIP/ /FLOP etc) deve ser transferido para o interior do programa.

Inicialmente, e a partir do esquema existente, vamos expor o exemplo tratado neste programa.

NUMERAMOS TODOS OS NÓS DOS DISPOSITIVOS — ENTRADA OU SAÍDA — SEQUENCIALMENTE; NÃO ESQUECENDO QUE OS NÓS LIGADOS ENTRE SI APENAS POSSUEM UM NÚMERO.

NESTE EXEMPLO TEREMOS UM NÚMERO DE NÓS =6.

Para o nosso exemplo, iremos considerar o número de impulsos de clock = 12.

No programa, a variável N corresponde a cada NÓ.

O elemento mínimo de contagem do tempo, é de 1/2 período de cada impulso do "clok".

A variável correspondente aos impulsos é T.

O "array" denominado A (N,T) irá guardar os valores da temporização de cada nó (todos os nós).

Tomemos como exemplo o dispositivo NAND que está na última posição do esquema.

a b c

Na tabela da verdade correspondente, as entradas estão classificadas como 'a' e 'b' e a saída é denominada 'c'. A saída é sempre '1', excepto quando as entradas são a = 1 e b = 1. Apenas quando os nós 4 e 5 são '1', a saída do nó 'b' é zero.

Cada dispositivo lógico (AND;NAND;OR;NOR;FLIP FLOP, etc.) pertencente ao circuito ou esquema deve ser transferido para o interior do programa.

No caso tratado — 2 circuitos Flip Flop tipo D e um tipo J–K e um circuito NAND, tivemos de transferir as condições lógicas dos circuitos, para o programa:

1 FFlop D ; Linha 2020; linha 2030
2 FFlop D ; Linha 2040; linha 2050
3 FFlop J–K; Linha 2060; linha 2070; linha 2080
NAND; linha 2090; linha 2100

Note que algumas das instruções derivadas de cada circuito lógico, conforme pode ver na Figura, podem ser ignoradas, dependendo da configuração do circuito.

No presente caso, ambos os bi-estáveis devem ser inicialmente colocados a zero (RESET). A instrução correspondente à saída Q (negada) não é incluida no programa devido a não ser usada.

Das 5 instruções da figura flip-flop J–K serão entretanto reduzidas a 3 linhas (2060 a 2080).

Desde que os Flip-Flop sejam dependentes do clock (tempo), requerem mais atenção, como p. ex. o tipo D, caracterizado pela entrada do clock 'a'; entrada de dados 'b'; pre-set 'e'; clear 'f'.

(Cont. no próximo número)

MICRODRIVE

SPECTRUM

INTERFACE

(Continuação)

O interface é o dispositivo que permite ao SPECTRUM comunicar com os periféricos de entrada/saída, do tipo impressora ou Microdrive.

O interface que ficará ligado ao SPECTRUM será do tipo RS-232 e será denominado INTERFACE 1.

É de notar que já foi lançado em Inglaterra o Interface 2 que será usado para ligação de JOYSTICK's e programas em ROM.

O INTERFACE RS-232 estabelece a comunicação entre dispositivos diferentes e, no caso da SINCLAIR, tem uma ficha de 9 pinos, dos quais dois são para recepção de sinais e outros dois para remeter dados.

Os dados são transferidos entre o SPECTRUM e o MICRODRIVE no modo SÉRIE, ou seja, cada *bit* segue sequencialmente através da linha de dados; por isso, teremos um outro dado em jogo, que é a velocidade de transmissão ou BAUD RATE, com valores entre 50 e 19200 bits por segundo.

DESCRITIVO DO MICRODRIVE

Trata-se de uma caixa c/ 90 * 85 * 40 mm com uma abertura na parte da frente, para introdução do cartucho.

Possui uma ficha do lado direito, onde é ligado um cabo flexível, que estabelece a comunicação com o INTERFACE 1.

No seu interior existem dois circuitos impressos. O que está situado na posição inferior, é onde se estabelecem as duas ligações para o exterior.

O circuito central (vertical), possui uma memória ROM e a cabeça de reprodução/gravação, a qual possui as duas funções: GRAVAR/LIMPAR.

Quando o microdrive está em funcionamento, um indicador (LED) fica luminoso.

- NÃO DEVE NUNCA, REMOVER O CARTUCHO, QUANDO O "LED" ESTÁ LUMINOSO.
- RETIRE O CARTUCHO QUANDO LIGAR OU DESLIGAR A ALIMENTAÇÃO.

O cartucho possui as dimensões de 45 * 35 * 7 mm e possui protecção contra gravação. Essa protecção deve ser retirada, quando pretendemos efectuar qualquer gravação.

Um cartucho contém uma só fita, de 5 metros de comprimento e com a largura de 1,5 mm. Trata-se de uma peça de fita girando em contínuo.

USAR O MICRODRIVE

Podemos usar o *Microdrive* após ter sido feita a ligação do *Interface 1* ao SPECTRUM (a alimentação será a do Spectrum).

Para utilização do dispositivo, necessitamos de um cartucho ou *Floppy Tape*. Voltamos a referir que não se trata de uma *diskette* vulgar, mas sim de uma fita, mecanicamente muito rápida.

A primeira operação será a de FORMATAR o cartucho que está virgem. A formatação dura cerca de 30 segundos, durante os quais o

rectângulo do ecran (conhecido como *Border*) ficará em acção de *flash*. No final teremos uma mensagem de OK.
A acção de formatar (ou inicializar) é fundamentalmente um conjunto de operações de definição de áreas, onde o computador pode ler ou escrever e também demarcar as áreas que não podem ser usadas.
A repetição desta operação leva à destruição de todo o conteúdo da fita, e só é efectuada quando queremos usar uma fita cujos dados já não têm interesse para nós.
A instrução que permite a inicialização da fita tem o nome de FORMAT e será usada com o *microdrive*, com o *interface* RS-232 e com a rede local *(Network)*.

FORMAT — "m"; 1; "TESTE"
FORMAT — instrução
"m"; 1 — microdrive 1
"Teste" — nome que pode ter até 10 caracteres

OUTRAS INSTRUÇÕES:

- CAT 1 — será a instrução usada para obter a lista de todos os nomes de programas ou conjuntos de dados que gravou na fita.
- SAVE * "m"; 1; "TESTE" — destina-se a gravar um programa na fita que está no *Microdrive* 1. O asterisco indica ao computador que o SAVE se destina ao *microdrive* e não ao gravador. "m"; 1 significa *microdrive* 1 e "TESTE" será o nome do programa. Um cartucho pode estar protegido para que não se possa escrever na fita, se lhe for retirada uma pequena peça plástica.
- VERIFY * "m"; 1; "TESTE" — verifica se o programa foi gravado anteriormente.
- LOAD * "m"; 1; "TESTE" — executa a reprodução de um programa gravado na fita.
- MERGE — permite juntar 2 programas num só. Pode ser usada desde que nenhum dos programas tenha sido gravado com SAVE ...... LINE número.

Uma das inovações significativas é poder-se preparar um programa com AUTO-RUN, ou seja, ligar o computador e arrancar com um determinado programa.

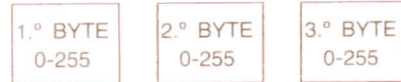
IMPORTANTE! — O comando RUN do SPECTRUM deixa de funcionar e passa a escrever-se R . . . U . . . N, letra a letra.

O programa será gravado sob a forma:
SAVE * "m"; 1; "RUN" LINE 100
(ou outro número)

Entrando a palavra RUN, a partir do teclado, o programa será reproduzido corrido a partir da linha 100.

FLOPPY TAPE (Cartucho de Fita)

1.° BYTE 0-255	2.° BYTE 0-255	3.° BYTE 0-255

← direcção da fita

No desenho acima pretendemos representar a forma como ficam gravados os *bits* e os *bytes* (BYTE = 8 BITS) ao longo de uma fita.
Todos os dados são gravados na fita, em duas pistas *(track)*, e os *bits* (um *bit* é a unidade de informação — 0 ou 1) são dispostos alternadamente em cada uma das pistas.
Em princípio não nos devemos preocupar se o facto de o *Microdrive* ter duas cabeças o torna mais lento ou sobre o comprimento que ocupa um *bit* ao longo de uma fita.
O bloco de DADOS *(data)* inicia-se com uma frente de 12 *bytes*, dos quais 10 são zeros e dois são "255" (255 em hexadecimal são FF, e em binário são ////////) — isto destina-se a um perfeito reconhecimento do início de um bloco de dados.

MICRODRIVES

RESPOSTAS A ALGUMAS QUESTÕES

PODEMOS USAR OS PROGRAMAS TRADICIONAIS?

Em primeiro, a maioria dos programas já existentes funcionam; apenas nos casos em que esses programas fazem uso do "interrupt control", os programas não funcionarão!
De qualquer modo, se o programa já existir quando obteve o microdrive, não tente usar o microdrive ou a rede externa, mas use o programa exactamente como se o microdrive não existisse.

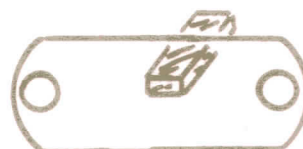
QUAL O NÚMERO MÁXIMO DE FILES QUE PODEMOS GRAVAR NUM CARTUCHO?

Cada file ocupa um mínimo de 512 Bytes ou seja um "buffer" ou tampão.
O catálogo ou directório das files mostra um número máximo de 50 nomes de files, embora o excesso, permaneça na fita e possa ser chamado em qualquer altura.

QUAL O TEMPO MÍNIMO PARA GRAVAR UMA FILE?

Será preferível falar em tempo médio, que no caso de Files de curta capacidade, será cerca de 10 segundos.

QUAL O TEMPO MÉDIO PARA ENCONTRAR UMA FILE ARQUIVADA NUM CARTUCHO?

O sistema usado pelo microdrive, é descrito desta forma:

Não existe enrolamento da fita mas sim uma passagem em contínuo.
Se a ficha (file) que pretendemos ler se encontra para além da cabeça de leitura, teremos de esperar que a zona da fita se situe junto da cabeça de leitura. Deste modo teremos entre 1 a 7 segundos para encontrar uma ficha!

CLUBE Z80

INSCRIÇÃO COMO ASSOCIADO

O **CLUBE Z80** está aberto a todos os utilizadores de microcomputadores.

A intenção de associar os entusiastas das micro-máquinas, é exclusivamente a de permitir:

1 — PUBLICAÇÃO DE UM JORNAL MENSAL, onde sejam publicados programas de uso geral ou específico como no caso da educação.

2 — PROMOVER TROCAS DE PROGRAMAS, e trocas de experiências; tanto no caso do Software (programação), como no caso do Hardware (electrónica).

3 — PROMOVER DESCONTOS NA AQUISIÇÃO DE PROGRAMAS.

4 — LANÇAR CURSOS DE PROGRAMAÇÃO EM BASIC — PASCAL OU OUTRAS LINGUAGENS E DIVULGAR O USO DE LINGUAGEM MÁQUINA.

NOME ....

IDADE .... COMPUTADOR TIPO ....

PROFISSÃO ....

ENDEREÇO ....

.... TELEF. ....

ASSINATURA ANUAL — Esc. 1 500$00 ☐

ASSINATURA SEMESTRAL — Esc. 750$00 ☐

CHEQUE OU VALE DO CORREIO

N.º ....

BANCO ....

DATA ....../....../......

JÁ SÓCIO ☐

NOVO SÓCIO ☐ → A partir do mês de .... (inclusive)